

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-051962

(43)Date of publication of application : 23.02.2001

(51)Int.Cl.

G06F 15/177

(21)Application number : 11-229355

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 13.08.1999

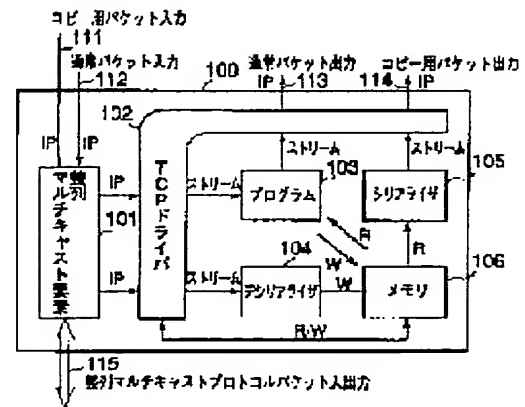
(72)Inventor : ENDO KOTARO

## (54) PROGRAM MULTIPLEXING AND EXPANDING METHOD

## (57)Abstract:

PROBLEM TO BE SOLVED: To provide a method for multiplexing a final program through sorting multicast regardless of the state of a computer.

SOLUTION: This is the method for multiplexing a final program 103 to be run on plural computers composing a distributed system through sorting multicast 101. In this case, a process is newly generated separately from a program under running composed of the sorting multicast 101, the final program 103 and the state of computers. Next, the state of a computer under running the process is copied from the process under running through the sorting multicast 101 to the new process.



## LEGAL STATUS

[Date of request for examination]

15.09.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号  
特開2001-51962  
(P2001-51962A)

(43) 公開日 平成13年2月23日 (2001.2.23)

(51) Int.Cl.<sup>7</sup>

G 0 6 F 15/177

識別記号

6 7 8

F 1

G 0 6 F 15/177

テーマコード(参考)

6 7 8 B 5 B 0 4 5

審査請求 未請求 請求項の数 4 O L (全 7 頁)

(21) 出願番号

特願平11-229355

(22) 出願日

平成11年8月13日 (1999.8.13)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 遠藤 浩太郎

東京都府中市東芝町1番地 株式会社東芝  
府中工場内

(74) 代理人 100058479

弁理士 鈴江 武彦 (外6名)

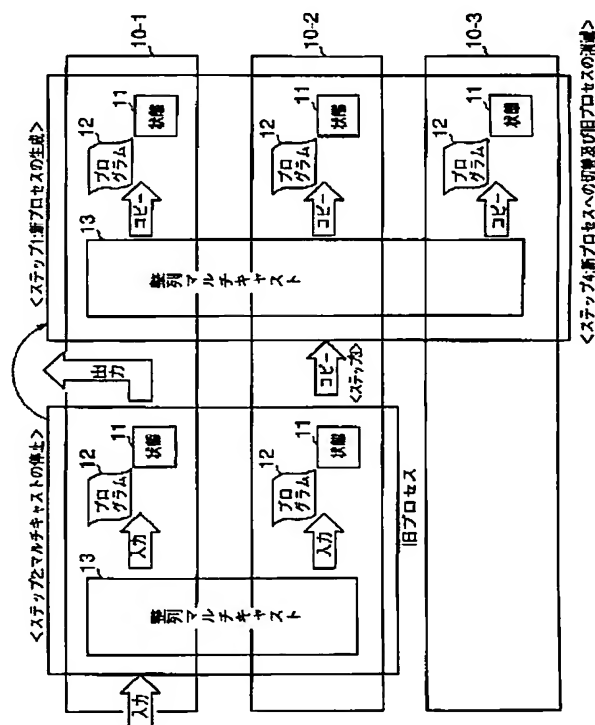
Fターム(参考) 5B045 BB04 BB49 JJ43

(54) 【発明の名称】 プログラム多重化拡大方法

(57) 【要約】

【課題】 コンピュータの状態の如何に拘わらず、整列マルチキャストにより決定的プログラムを多重化する方法を提供すること。

【解決手段】 分散システムを構成する複数のコンピュータ204上で実行される決定的プログラム103を整列マルチキャスト101により多重化する方法であり、整列マルチキャスト101と決定的プログラム103とコンピュータ204の状態とから構成される実行中のプロセスとは別に、新規にプロセスを生成し、次いで、プロセスを実行中のコンピュータ204の状態を、実行中のプロセスから新規のプロセスへと整列マルチキャスト101を通してコピーする。



## 【特許請求の範囲】

【請求項1】 分散システムを構成する複数のコンピュータ上で実行される決定的プログラムを整列マルチキャストにより多重化する方法において、前記整列マルチキャストと前記決定的プログラムと前記コンピュータの状態とから構成される実行中のプロセスとは別に、新規にプロセスを生成し、次いで、前記プロセスを実行中のコンピュータの状態を、前記実行中のプロセスから前記新規のプロセスへと前記整列マルチキャストを通してコピーすることとを特徴とするプログラム多重化拡大方法。

【請求項2】 前記コンピュータの状態をコピーするに先だって、コンピュータの状態の整理を行なうことを特徴とする請求項1記載のプログラム多重化拡大方法。

【請求項3】 前記コンピュータの状態の整理は、不要情報の削除、メモリ配置の最適化及びオブジェクト固有の最適化のうち少なくとも一つであることを特徴とする請求項2記載のプログラム多重化拡大方法。

【請求項4】 前記新規に生成するプロセスを、前記実行中のプロセスが存在するコンピュータ群とは別のコンピュータ群上に生成することを特徴とする請求項1乃至請求項3のいずれか一項記載のプログラム多重化拡大方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、複数のコンピュータがネットワークで接続された分散システムにおいて決定的プログラムを整列マルチキャストを用いて多重化する方法に関する。

## 【0002】

【従来の技術】先ず、本明細書中で用いられる決定的プログラム、整列マルチキャスト及び多重化の意義について説明する。

【0003】決定的プログラムは次のように説明される。すなわち、図6に示すように、決定的プログラム12は、コンピュータ10に入力が与えられると、その時のコンピュータ10の状態11によって、出力と、次の状態を決めるプログラムであると考えることができる。つまり、決定的プログラム12は、入力が決まれば、次の状態12と出力とが一意的に決まるプログラムである。具体的には、不定値や乱数の参照がないプログラムのことを言う。

【0004】図7に示すように、決定的プログラムの特徴は、初期状態と入力列が決まれば、その動作が一意的であることである。以下、プログラムと称するとき、決定的プログラムを言う。

【0005】また、整列マルチキャストは、次のように説明される。すなわち、複数のコンピュータが結合した分散システムのような環境では、それぞれのコンピュータが独立して動作する。従って、これらのコンピュータ

を同期的に動作させる場合には、特別な仕組みが必要になってくる。整列マルチキャストは、それぞれのコンピュータから全てのコンピュータにデータを配送するプロトコルであり、データの到着順序が全てのコンピュータで同じであることを保証するものである。

【0006】整列マルチキャストを、図8を参照して具体例で説明する。コンピュータ10-2から時刻t20に送信されたデータAは、図示しない整列マルチキャストにより全てコンピュータ10-1、10-2、10-3に時刻t11、t21、t31に受信される。コンピュータ10-3から時刻t30に送信されたデータBは、整列マルチキャストにより全てのコンピュータ10-1、10-2、10-3に時刻t12、t22、t32に受信される。この場合、各コンピュータ10-1、10-2、10-3はデータAとデータBとが受信されるが、整列マルチキャストはこの2つの受信順序が全てのコンピュータ10-1、10-2、10-3で同一になるように制御する。

【0007】さらに、多重化は、次のように説明される。すなわち、分散システムでは、それぞれのコンピュータが独立に故障する可能性がある。仮に、一つのコンピュータが故障しただけで、システム全体が機能しない場合は、分散システムの稼働率は一台のコンピュータの稼働率よりも低くなってしまう。

【0008】かかる事態を防止するために、システム全体に係わる処理は多重化することが必要である。逆に、多重化することによって、分散システムの稼働率を一台のコンピュータの稼働率よりも高くすることが可能である。例えば、稼働率99パーセントのコンピュータ10台で構成する分散システムが、全く多重化されていないとすると、その分散システムの稼働率は90パーセント程度である。また、稼働率99パーセントのコンピュータ2台に多重化された処理の稼働率は99.99パーセント程度である。

【0009】次に、図9を参照して、整列マルチキャストを用いた多重化法について説明する。この例は、コンピュータ10-1、10-2、10-3を有する分散システムにおいて、プログラムの実行を、整列マルチキャストを用いて多重化する例である。

【0010】図9に示すように、まず、全てのコンピュータ10-1、10-2、10-3はある決まった初期状態11、例えば全ての変数が0、から始まる。入力されるデータは、必ず整列マルチキャスト13を通して全てのコンピュータ10-1、10-2、10-3に配送され、夫々のプログラム12に入力される。出力は一つのコンピュータの出力をとるようにする(図9ではコンピュータ10-1である)。各プログラムの入力列は、整列マルチキャスト13により同一順序になっているので、プログラムの特徴により、全てのコンピュータ10-1、10-2、10-3の状態11が同一に保たれ、

出力列も全て同じになる。つまり、プログラムの実行が多重化される。

【0011】つぎに、整列マルチキャストを用いて多重化する方式と、マスタースレーブ方式で多重化する方式との相違について説明する。すなわち、マスタースレーブ方式は、マスタと呼ばれるコンピュータ上でプログラムを実行しながら、定期的にスレーブと呼ばれるコンピュータに状態を転送しておき、マスタが故障停止したときスレーブ側でのプログラムの実行に切り替えることにより多重化を実現する。

【0012】しかし、マスタースレーブ方式では、切り替えの際に後戻りが発生するため、コンピュータの故障停止の際の切り替え処理が複雑になり、時間がかかるという問題がある。

【0013】一方、整列マルチキャストを用いる場合は、コンピュータの故障停止の際に、後戻りがないため、切り替え処理は単純であり、時間がかからない。

【0014】また、マスタースレーブ方式では、定期的に状態をコピーするためのオーバーヘッドが発生するが、整列マルチキャストを用いる場合、このオーバーヘッドは

【0015】このように、システム全体の信頼性、性能に係わる処理は、整列マルチキャストによって多重化することが好ましい。

【0016】一方、マスタースレーブ方式は、非決定的プログラムや、スレーブ側でプログラムを実行することが好ましくない場合に適している。

【0017】上記の整列マルチキャストによる多重化の方法では、はじめからコンピュータが全て稼働していることを前提としていた。しかし、実際には途中から多重化を始めなければならない場合もある。例えば、故障していたコンピュータが復旧した場合や、新たにコンピュータを追加した場合がある。これらの場合には多重化の拡大が必要である。

【0018】かかる整列マルチキャストによる多重化を拡大する従来例を、図10を参照して説明する。図10において、ステップ1として、整列マルチキャスト13を一時停止する。次に、ステップ2として、組み込まれるコンピュータ10-3に、状態11をコピーする。次に、ステップ3として、整列マルチキャスト13のグループを拡大し、再開する。

【0019】この方式では、仮に、状態のコピーが正確に行なわれなかったとすると、組み込まれたコンピュータの動作が、他のコンピュータと異なってしまう。もちろん、整列マルチキャストを一時停止することにより、状態を不変にした上でコピーしているので、このようなことは原理的には起きず、整列マルチキャストを再開したときに全てコンピュータが同一の動作を始まるようになっている。

【0020】

【発明が解決しようとする課題】しかしながら、コンピュータの状態が複雑な場合には、状態を正確に取得することは容易ではなく、しばしばバグが潜んでいる場合がある。そのような場合に、従来方式では、組み込まれたコンピュータだけが誤動作し、他のコンピュータは正常に動作するため、却って不具合が発見されにくくなるという問題がある。

【0021】本発明の目的は、コンピュータの状態の如何に拘わらず、整列マルチキャストにより決定的プログラムを多重化することが可能な方法を提供することにある。

【0022】

【課題を解決するための手段】かかる課題を解決するために本発明は、分散システムを構成する複数のコンピュータ上で実行される決定的プログラムを整列マルチキャストにより多重化する方法において、前記整列マルチキャストと前記決定的プログラムと前記コンピュータの状態とのうち少なくとも一つから構成される実行中のプロセスとは別に、新規にプロセスを生成し、次いで、前記プロセスを実行中のコンピュータの状態を、前記実行中のプロセスから前記新規のプロセスへと前記整列マルチキャストを通してコピーすることを特徴とする。

【0023】本発明によれば、実行中のプロセスを拡大するのではなく、別にプロセスを新しく生成し、実行中のプロセスから新規のプロセスへ状態をコピーするのであるから、状態取得の不具合等の影響は多重化そのものに影響受けることが無い。また、従来のように整列マルチキャスト自体の拡大を必要としなく、よって整列マルチキャストプロトコルを従来よりも簡素化することができ、性能、信頼性の向上につながる。

【0024】

【発明の実施の形態】以下本発明にかかるプログラム多重化拡大方法の一実施形態を図面を参照して説明する。まず、実施形態の説明に先立ち本発明で用いられるプロセスの概念を説明する。即ち、プロセスとは、整列マルチキャストと、各コンピュータによって多重化されたプログラムと、状態とで構成され、稼働率の高い実行単位である。

【0025】前述のプロセスの概念の下で、本実施形態の多重化方法は、従来方式のように、実行中のプロセスを拡大するのではなく、別にプロセスを新しく生成し、実行中のプロセスから新規のプロセスへ、状態をコピーするものである。

【0026】従って、本実施形態の多重化方法は、図1に示すように、ステップ1として、新規のプロセスを別に生成する。ステップ2として、実行中のプロセスの整列マルチキャストを一時停止する。ステップ3として、実行中のプロセスから、新規のプロセスに状態をコピーする。ステップ4として、新規のプロセスが入出力を行なうように切り替えられ、旧プロセスは消滅する。この

場合、新規のプロセスへの状態のコピーは、整列マルチキャストを通して行なわれるので、全てのコンピュータで同一になることが保証される。また、仮に、状態を正確に取得することができなかったとしても、全てのコンピュータの状態は同一である。つまり、この場合には全てのコンピュータが、再開後に同じ誤動作を始める。

【0027】次に、図2、図3を参照して、本実施形態のプログラム多重化拡大方法を、該方法が適用される分散システムとの関係で説明する。図2、図3は、本実施形態のプログラム多重化拡大方法が適用される分散システムを示しており、ネットワーク202に複数のコンピュータ204が接続されている。各コンピュータ204は、少なくとも、複数のプロセス要素100及び仮想アドレスフィルタ201を有する。

【0028】ここで、プロセス要素100は複数有り、これらは各コンピュータ204上に設けられており、これら集合がプロセスを形成する。このプロセス要素100は、図3に示すように、整列マルチキャスト要素101、TCP（トランスミッション・コントロール・プロトコル）ドライバ102、プログラム103、デシリアライザ104、シリアライザ105で構成される。

【0029】図3において、整列マルチキャスト要素101は、各コンピュータ204上に設けられており、それらの集合が整列マルチキャストを形成する。整列マルチキャスト要素101間では、整列マルチキャストプロトコルパケット入出力115の受け渡しが行われ、整列マルチキャストとしての機能を実現する。

【0030】TCPドライバ102は、パケットの再送、並び変えなどを制御し、ランダム順序なパケットデータをシーケンシャルデータに変換する。通常パケットを処理するTCPの状態は、メモリ106に保持される。コピーパケットを処理するTCPの状態は、メモリ106に保持されない。

【0031】プログラム103は、任意のプログラムであり、TCPドライバ102からシーケンシャルデータを入力し、メモリ106を参照、更新し、TCPドライバ102にデータを出力する。プログラム103は、全てのコンピュータ上で同一である。

【0032】デシリアライザ104は、TCPドライバ102から入力するシーケンシャルデータをメモリ106の状態に変換し書き込む。

【0033】シリアライザ105は、メモリ106の状態をシーケンシャルデータに変換しTCPドライバ102に出力する。

【0034】メモリ106は、状態を保持する。

【0035】コピー用パケット入力111は、コピー用パケットを整列マルチキャスト要素101を通して整列化し、TCPドライバ102に入力する。

【0036】通常用パケット入力112は、通常パケットを整列マルチキャスト要素101を通して整列化し、

TCPドライバ102に入力する。

【0037】コピー用パケット出力113は、シリアライザ105から出力されたシーケンシャルデータがTCPドライバ102で変換されたものである。

【0038】通常パケット出力114は、プログラム103から出力されたシーケンシャルデータがTCPドライバ102で変換されたものである。

【0039】整列マルチキャストプロトコルパケット入出力115は、コピー用パケット入力101が整列マルチキャストプロトコルを実現するために行う。

【0040】仮想アドレスフィルタ201は、プロセスに対応する論理IPアドレスを持っている場合にのみ、コピー用パケット入力111、通常用パケット入力112、コピー用パケット出力113、通常パケット出力114を通過させる。プロセスに対応する論理IPアドレスは、どれか一つのコンピュータ上にしか存在しないので、プロセスを構成するプロセス要素のうち、一つプロセス要素のみがネットワークに入出力を行なうことになる。

【0041】ネットワーク202は、コピー用パケット及び通常パケットが流れ、インターネットに接続するLANである。

【0042】インタコネクタ203は、整列マルチキャストプロトコルパケットがながれる専用の高速ネットワークである。

【0043】コンピュータ204は、複数台で分散システムを構成する。図2では2台だけ示しているが、2台を超える場合も含まれる。

【0044】論理IPアドレス205は、一つのプロセスに対応する。実行中のプロセスに対応する通常の論理IPアドレスと、拡大の時にのみ使われる立ち上げ論理IPアドレスがある。

【0045】尚、IP（インターネットプロトコル）はIPパケットを示し、ストリームはシーケンシャルデータを示し、Rはメモリ106からの読み出しを示し、Wはメモリへの書き込みを示し、R/Wはメモリへの読み出し／書き込みを示している。

【0046】次に、図4及び図5を参照して本実施形態における多重化の拡大の動作を説明する。尚、図4は拡大中の動作を示す、図5は拡大後の動作を示している。

【0047】図4の矢印で示すように、本発明では拡大に際して、新規のプロセスを生成する（ステップ1）。このとき、実行中の整列マルチキャストは一時停止される（ステップ2）。新規のプロセスには、立ち上げ論理IPアドレス205Bが割り振られ、実行中のプロセス（旧プロセス）に割り当てられている通常論理アドレス205Aと区別される。ステップ3として、状態のコピーは、メモリ106→シリアライザ105→仮想アドレスフィルタ201→整列マルチキャスト要素101→TCPドライバ102→デシリアライザ104→メモリ1

06を通して行なわれる。

【0048】図5は、状態のコピーが終わった状態を示しており、ステップ4として、旧プロセスは消滅し、旧プロセスに対応していた通常論理アドレス205Aは、新プロセスに対応するように切り替えられる。同時に新プロセスに対応していた立ち上げ論理IPアドレス205Bは消滅する。

【0049】このように本実施形態では、多重化そのものの仕組みが状態取得の不具合などによって破綻しにくいという特徴を有しており、整列マルチキャスト自体の拡大は必要ない。従って、整列マルチキャストプロトコルを、従来よりも簡素化することができ、性能及び信頼性が向上する。また、多重化そのものの仕組みが状態取得の不具合などによって破綻しにくいいため、システム全体の信頼性の向上、開発期間の短縮化に寄与する。

【0050】本発明は上述した実施形態に限定されない。すなわち、前述の実施形態において、103プログラムをJava（米国Sun Microsystems社が開発したプログラミング言語）プログラムとし、104、105をそれぞれJava上でのデシリアライザ、シリアライザとすることができる。

【0051】この場合、シリアライズの機能をJava自身が持っているため、新たにJava上のデシリアライザ104、シリアライザ105を作成する必要がないため開発期間が短縮できる効果を有する。

【0052】また、前述の実施形態において、プログラム103をリスプインタプリタ（Lisp Interpreter）を実装するプログラムとし、シリアライザ105を、使用中のセルのみを出力するシリアライザとすることができる。この場合、プログラム103をリスプインタプリタに代えてJavaインタプリタとすることができる。

【0053】このようにプログラム103をリスプインタプリタ又はJavaインタプリタとし、シリアライザ105を使用中のセルのみを出力するものとした場合、旧プロセスと新プロセスとのメモリの内容は正確には同一にはならないが、実質的にはガベージ・コレクション（garbage collection）を行ったのと同じことになるので、実質的な動作は正常に保たれ、しかもメモリのフラグメンテーション（fragmentation）が解消されるなど性能面でも改善に寄与する。

【0054】この方式は、コンピュータの状態を正確に取得することができなかったとしても、全てのコンピュータの状態は同一に保たれるという特徴を積極的に利用したものである。

【0055】さらに、前述の実施形態において、新プロセスを、実行中のプロセスとは別のコンピュータ上に生成し、多重化の拡大を行うことなく、多重化の移動又は縮小を行うようにしても良い。この応用例は、従来、多重化の移動は容易ではなかったことを考慮すると、シス

テムのリプレースメントや、メンテナンスのためのコンピュータの一時停止などの際に有効である。

【0056】

【発明の効果】以上のように本発明によれば、実行中のプロセスとは別に、新規にプロセスを生成し、次いで、前記プロセスを実行中のコンピュータの状態を、前記実行中のプロセスから前記新規のプロセスへと前記整列マルチキャストを通してコピーすることにより、実行中のプロセスを拡大するのではなく、別にプロセスを新しく生成し、実行中のプロセスから新規のプロセスへ状態をコピーするのであるから、状態取得の不具合等の影響は多重化そのものに影響受けることが無い。また、従来のように整列マルチキャスト自体の拡大を必要としなく、よって整列マルチキャストプロトコルを従来よりも簡素化することができ、性能、信頼性の向上につながる。

【0057】よって、本発明によれば、コンピュータの状態の如何に拘わらず、整列マルチキャストにより決定的プログラムを多重化することが可能な方法を提供することができる。

【図面の簡単な説明】

【図1】本発明によるプログラム多重化拡大方法の一実施形態を示す図。

【図2】同実施形態の方法が適用される分散システムの一例を示すブロック図。

【図3】図2におけるプロセス要素の詳細ブロック図。

【図4】同実施形態における多重化を拡大中の動作を説明するブロック図。

【図5】同実施形態における多重化を拡大後の動作を説明するブロック図。

【図6】プログラムの概念を示す図。

【図7】決定的プログラムの状態遷移を示す図。

【図8】従来技術におけるイメージ入力システムのブロック図。

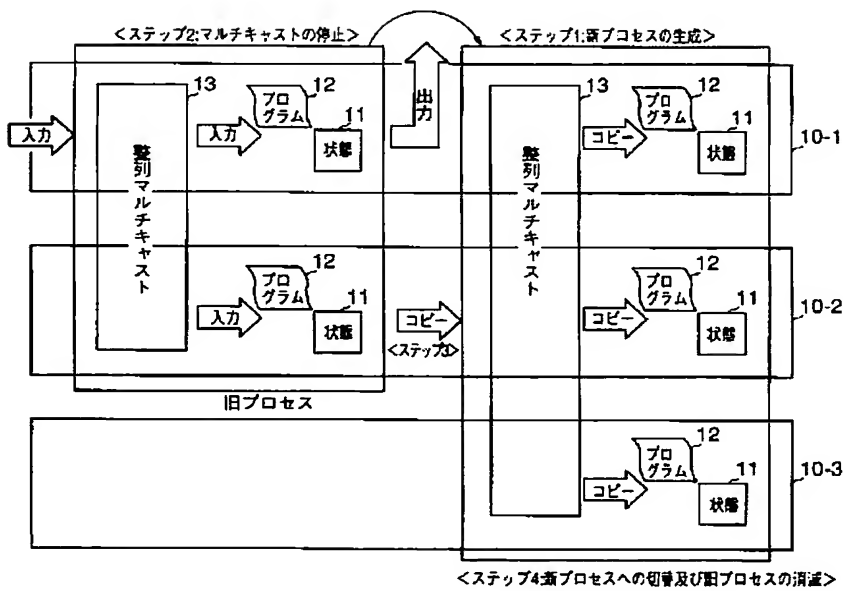
【図9】整列マルチキャストによるデータの配送の例を示す図。

【図10】整列マルチキャストによる多重化法の従来例を示す図。

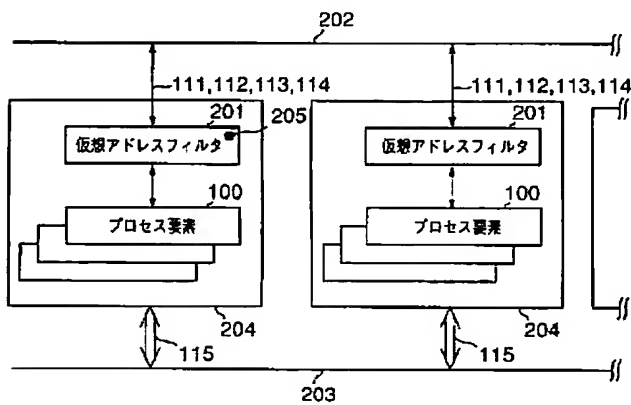
【符号の説明】

100…プロセス要素、101…整列マルチキャスト要素、102…TCPドライバ、103…プログラム、104…デシリアライザ、105…シリアライザ、106…メモリ、111…コピー用パケット入力、112…通常用パケット入力、113…コピー用パケット出力、114…通常パケット出力、115…整列マルチキャストプロトコルパケット入出力、201…仮想アドレスフィルタ、202…ネットワーク、203…インタコネク、204…コンピュータ、205…論理IPアドレス。

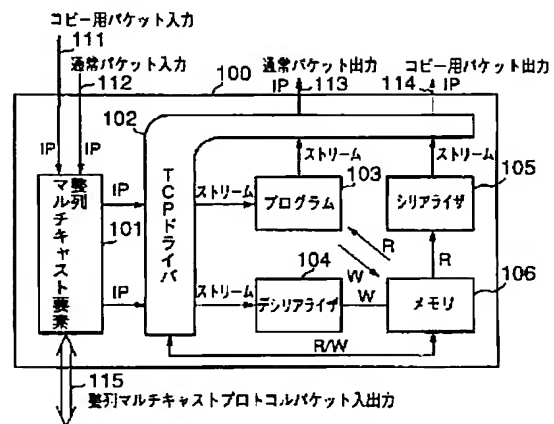
【図1】



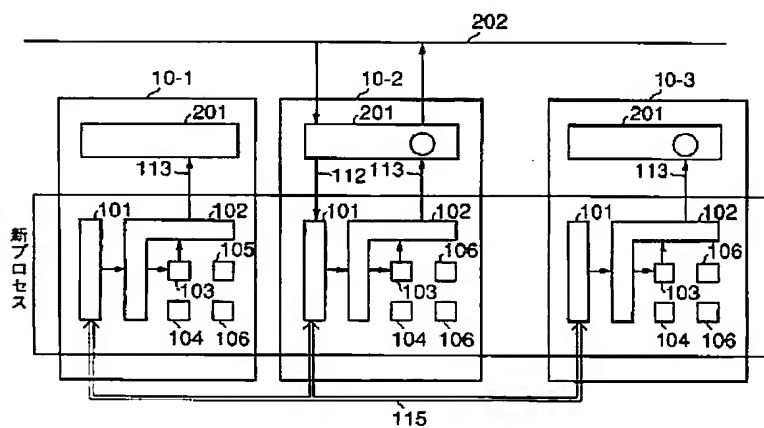
【図2】



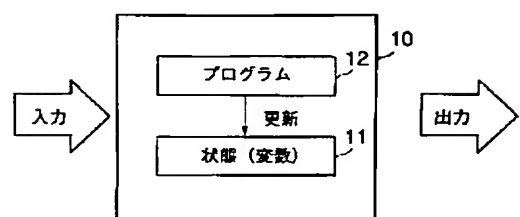
【図3】



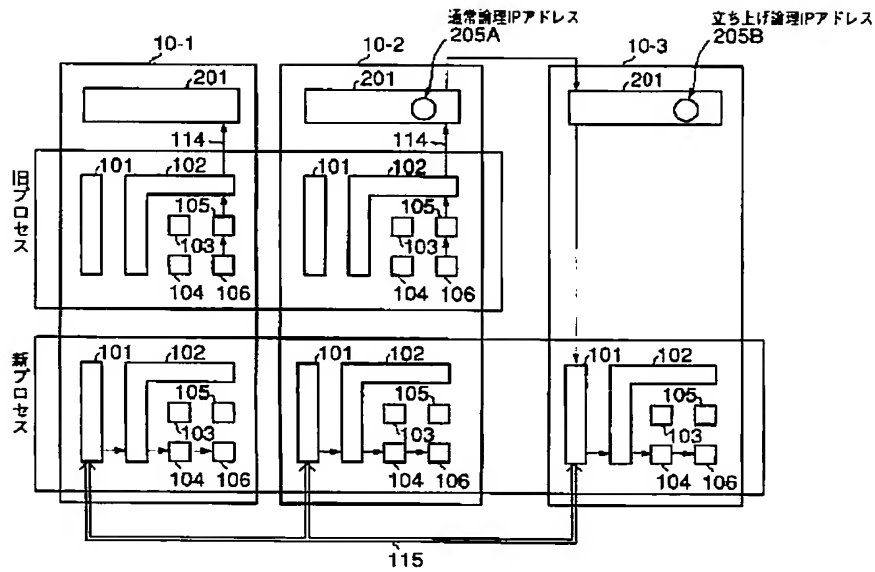
【図5】



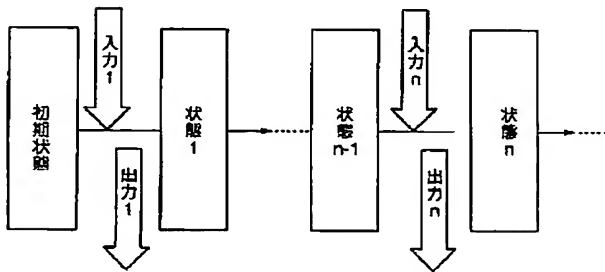
【図6】



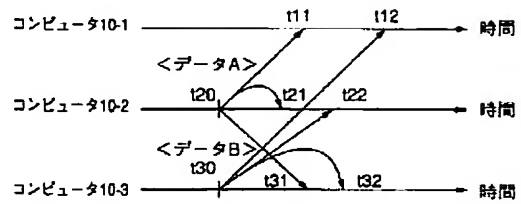
【図4】



【図7】



【図8】



【図10】

【図9】

